5th International Conference on Bioinformatics and Computational Biology 2013

(BICoB-2013)

Honolulu, Hawaii, USA
4-6 March 2013

ISBN: 978-1-62276-971-1

5th International Conference on Bioinformatics and Computational Biology 2013

(BICoB-2013)

Honolulu, Hawaii, USA
4-6 March 2013

ISBN: 978-1-62276-971-1

Printed from e-media with permission by:

Curran Associates, Inc.
57 Morehouse Lane
Red Hook, NY 12571

Some format issues inherent in the e-media version may also appear in this print version.

Original ISBN: 978-1-880843-89-5 (Out of Print)
Reprint ISBN: 978-1-62276-971-1

Printed by Curran Associates, Inc. (2013)

International Society for Computers and Their Applications
975 Walnut Street, Suite 132
Cary, NC 27511-4216

Phone: (919) 467-5559
Fax: (919) 467-3430

isca@ipass.net

Additional copies of this publication are available from:

Curran Associates, Inc.
57 Morehouse Lane
Red Hook, NY 12571 USA
Phone: 845-758-0400
Fax: 845-758-2634
Email: curran@proceedings.com
Web: www.proceedings.com

International Program Committee

Conference Chairs
Hisham Al-Mubaid
University of Houston – Clear Lake, USA
Reda Alhajj
University of Calgary, Canada

Program Chairs
Fahad Saeed
NHLBI, National Institutes of Health (NIH), USA
Bhaskar DasGupta
University of Illinois at Chicago, USA

Mukul Bansal	Massachusetts Institute of Technology
Wenk Carola	Tulane University
Kun-Mao Chao	National Taiwan University
Ping Chen	University of Houston
Jake Chen	IUPUI
Peter Clote	Boston College
Scott Emrich	University of Notre Dame
Oliver Eulenstein	Iowa State University
Nicholas Flann	Utah State University
Osamu Gotoh	Kyoto University
Thomas R. Ioerger	Texas A&M University
Minghui Jiang	Utah State University
Tamer Kahveci	University of Florida
Ming-Yang Kao	Northwestern University
Marek Karpinski	University of Bonn
Marsolo Keith	Cincinnati Children's
Hans Armin Kestler	Universitat Ulm
Danny Krizanc	Wesleyan University
Yaohang Li	Old Dominion University
Yunkai Liu	Gannon University
Keith Marsolo	University of Cincinnati
Shinichi Morishita	University of Tokyo
Bogdan Pasaniuc	Harvard School of Public Health
Marc Pomplun	University of Massachusetts at Boston
Yann Ponty	Ecole Polytechnique
Ponraj Prabakaran	National Institutes of Health
Salari Raheleh	Stanford University
Huzefa Rangwala	George Mason University
Mireille Regnier	INRIA
Kay Robbins	University of Texas at San Antonio
Jianhua Ruan	University of Texas at San Antonio
Walter Ruzzo	University of Washington
Rajasekaran Sanguthevar	University of Connecticut
Sahar Al Seesi	University of Connecticut
Bin Song	University of Florida
Wing-Kin Sung	National University Singapore
Sing-Hoi Sze	Texas A&M University
Ugo Vaccaro	Universita di Salerno
Jianxin Wang	Central South University
Li-San Wang	University of Pennsylvania
Tiffani Williams	Texas A&M University
Damian Wojtowicz	National Institutes of Health
Yufeng Wu	University of Connecticut
Dong Xu	University of Missouri
Jinbo Xu	Toyota Technological Institute at Chicago

Hui Yang	San Francisco State University
Xiao Yang	Broad Institute
Kim Yoo-Ah	National Institutes of Health
Kun Huang	Ohio State University
Erliang Zeng	University of Notre Dame
Patrick Zhao	Noble Foundation

Message from Program Chairs

As the program co-chairs of the Fifth International Conference on Bioinformatics and Computational Biology, BICOB-2013, and on behalf of the organizing committee, we would like to extend our thanks to all the contributors and participants of this conference. This year, the BICOB conference is held in Honolulu, Hawaii, March 4 – 6, 2013. We would like to welcome all the participants to Honolulu. The fifth BICOB conference offers the opportunity to researchers and scientists, all over the world, to present and discuss their research results, techniques and findings with other researchers having similar interests in the fields of bioinformatics and computational biology. This year BICOB is held in Honolulu as it is considered one of the best destinations for entertainment, scientific meeting, and conventions, and with fabulous sight-seeing.

The BICOB-2013 conference features two keynote talks by well-known research scientists and 16 sessions of regular paper presentations. The conference addresses a broad range of topics in the bioinformatics area including: machine learning in bioinformatics, data mining applications in bioinformatics, gene and protein studies, microarray research and applications, diseases and drug related research, biological networks, regulatory networks, DNA and RNA research and more.

This year, the BICOB-2013 conference is gathering bioinformatics researchers, scientists, practitioners, and attendants from many countries (e.g., Brazil, China, Germany, Hong Kong, India, Iran, Japan, Lebanon, UK, and USA). The participants are coming from various research institutions like universities, corporations, and government research agencies (e.g. NIH).

Each paper submitted to BICOB-2013 was reviewed and evaluated by three reviewers who are usually members of the international program committee. Additionally, papers have been scanned by referees judging the originality, significance, technical contents, application contents and presentation style. We used the Microsoft Online Conference Management System to automate the workflow for submission and refereeing.

We gratefully acknowledge the professional work of the international program committee and the sub-reviewers contributing tremendous amount of referee reports. We owe great thanks to ISCA president and board of directors for the well-organized conference management. We also, would like to thank the general chairs of the conference Dr. Hisham Al-Mubaid and Dr. Reda Alhajj for their guidance, support, and work. We also want to thank all presenters and attendees for actively contributing to the success of BICOB-2013, and look forward to excellent presentations and fruitful discussions, which will for sure broaden our professional horizons. All participants are invited to make new friends within the BICOB family. We sincerely wish to every participant a very enjoyable and beneficial time at BICOB-2013.

With Best Regards

Fahad Saeed
Bhaskar DasGupta
Program Chairs, BICOB–2013

March 2013

Table of Contents

Towards Better Base Classifiers for Ensemble Classification of Gene Expression Data

William Duncan
Computer Science Division
Louisiana State University
Baton Rouge, LA 70803
duncan@csc.lsu.edu

Jian Zhang
Computer Science Division
Louisiana State University
Baton Rouge, LA 70803
zhang@csc.lsu.edu

Abstract

Classification using gene expression data has the potential to provide better cancer diagnosis. A main challenge in such classification is to train a good model using a (relatively) small number of examples, each involving a large number of genes. Ensemble of classifiers that are based on subsets of genes has been proposed to cope with this situation. Although a lot of works have explored different techniques to create gene subsets, few work has considered the impact of the type of the classifiers on the performance of the ensemble. Since different types of classifiers have different characteristics, they may lead to different performance. We investigated the effects of classifier flexibility and regularization of the classifiers on the classification accuracy of the ensemble. Our results suggest that rather than flexibility, regularization is a more important direction to explore for achieving higher accuracy. In particular, L1-regularization leads to the best performing ensemble. Interestingly, such performance advantage is obtained not by implicit feature selection but by reducing the influence of correlated genes.

Keywords: Ensemble Classification, L1 Regularization

1 Introduction

With the advance in DNA microarray technology, the expression levels of thousands of genes can be simultaneously measured in a single experiment. This opens a door to better diagnosis of cancers and other diseases. Diagnosis using gene expression data based on statistical and computational models has been subject to intensive researches in both the biomedical and the computer science community [1-6]. The diagnosis problem is essentially a classification problem: a classifier (model) can be trained using gene expression data obtained from groups of cancer/disease patients as well as normal persons. Once trained, the classifier can be used to predict (diagnosis) other people whose condition is unknown.

One of the main difficulties in classification using gene expression data is the small number of training examples v.s. the large number of features, i.e., genes. This leads to overfitting where the trained classifier fits to a few specific cases in a particular collection of training examples and cannot generalize to the other cases in the same class. For example, even though in reality, a gene (or a few genes) may not be relevant to a disease condition, the expression level of the gene(s) may appear as good indicator(s) for the condition if we look at only a very small number of examples (persons). A classifier trained using these examples will employ the gene(s), resulting in almost perfect classification for the training examples but very poor classification for other cases.

Ensemble method [7, 8] is an approach used to cope with this problem. Instead of training a single classifier, a collection of classifiers are trained and then combined to perform classification. The classifiers in the ensemble are called base classifiers. The base classifiers are often simpler. For example, they can be built upon a relatively small subset of genes. By focusing on a small subset of genes, we may avoid some of the "appear-to-be-good" genes. (In statistical terms, this lowers the variance.) On the other hand, because the subset of genes used may not be sufficient to capture the characteristic of the data fully, the individual base classifier may still not classify well. (In statistical terms, the base classifier has high bias.) The ensemble, by combining a collection of base classifiers, can take advantage from both sides, i.e., it is affected less by the "appear-to-be-good" genes and at the same time can be flexible enough to fit the data.

Clearly, an effective ensemble requires careful design to take full advantage from both sides. The design involves choice of techniques to generate the gene subsets, choice of base classifiers and choice to combine the output from the base classifiers to form the final prediction. Many works have looked into different techniques for creating gene subsets. Besides constructing subsets of genes by random selection [7, 8], researchers have investigated methods such as genetic algorithm [9, 10], partition based on Markov blanket [9, 11]. Also many classifiers have been used as base classifiers in an ensemble, e.g., tree classifier [12], nearest neighbor classifier (knn) [11], naïve Bayes classifier [11] and Support Vector Machine [9, 10] as a few examples.

Different types of classifiers have different characteristic. For example, tree classifier is more flexible than a linear classifier. (Consider a two-class classification. A linear classifier divides the space where the data lie into two regions by a hyperplane. On the other

hand, a tree classifier can divide the space into many regions.) Furthermore, classifiers with regularization perform implicit feature ranking that removes or reduces the effects of redundant genes. These characteristics may lead to different performances among ensembles that use different types of classifiers as their base classifiers. In this paper we investigate the effects of the choice of base classifiers on the ensemble and focus on the following two problems:

1. Would a flexible classifier serve as a better base classifier?
2. Would a classifier with regularization serve as a better base classifier?

We conducted experiments to evaluate 3 types of classifiers serving as base classifiers for an ensemble: 1) tree classifier, 2) L2- regularized linear classifier (SVM) and 3) L1-regularized linear classifier. Our results show that in most cases, the ensembles that use regularized (both L1 and L2) base classifiers perform better than the one using the tree classifier. In all the cases, the ensemble using L1-regularized classifier gives consistently the best performance. This suggests that to design a good ensemble, flexibility of the base classifier is not a main concern but regularization techniques should be considered. In particular, base classifiers with L1-regularization may lead to an ensemble of better classification. Furthermore, our results show that the better performance is not due to implicit gene selection but rather due to the small magnitude of the coefficients caused by the regularization.

2 Method

We investigate the impact of different types of base classifiers on the classification accuracy of an ensemble by comparing the performance of the ensembles using the following types of base classifiers: 1) tree classifier, 2) L2- regularized linear classifier and 3) L1-regularized linear classifier. We focused on the ensemble whose base classifiers are built on a small random subset of the genes. We give the details of the ensemble and the based classifiers in the following sections.

2.1 Ensemble of Classifiers Using Randomly Selected Subsets of Genes

Let $X = \{x^{(1)}, x^{(2)}, \ldots, x^{(N)}\}$ be the gene expression data for the N training examples. $x^{(i)}$ is a d-dimension vector containing gene expression values and d is the number of genes involved in the data. Let $Y = \{y^{(1)}, y^{(1)}, \ldots, y^{(N)}\}$ be the corresponding class labels of the training examples. We construct an ensemble using 3 steps:

1. From the set of d genes involved in the gene expression data, construct M subsets of genes. Each subset consists of k randomly selected genes.
2. Train M base classifiers. The ith base classifier is trained using the training examples restricted to the genes in the ith subset, i.e., each classifier is trained using $\{\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots, \hat{x}^{(N)}\}$ and labels $\{y^{(1)}, y^{(1)}, \ldots, y^{(N)}\}$ where $\hat{x}^{(i)}$ is the vector derived by taking the elements of $x^{(i)}$ that correspond to the genes in the gene subset used by the classifier.
3. To classify a new case, the classification results from each base classifier are pooled together and a final classification is made by majority voting. When there are multiple classes (e.g., disease condition 1, 2, etc), the case is classified as the one with the most votes.

We constructed multiple ensembles using the above approach. Different ensembles use different base classifiers.

2.2 Classification Tree Classifier

Classification tree is a type of simple and widely used classifier. A classification tree is constructed by recursively growing nodes that partition the training examples into groups with better and better impurity measure. We used Gini index as the impurity measure for our tree construction.

One important difference between the tree classifier and a linear classifier is that they build different types of decision boundaries. A decision boundary is the boundary where points on one side belong to one class and points on the other belong to the other. A tree classifier partitions the space of the data points into multiple regions and assigns a class label to each region. The regions are delimited by multiple (axis-parallel) planes. On the other hand, the decision boundary of a linear classifier is a hyperplane in the data space. Thus, tree classifiers are more flexible and may be able to deal with situations where the data points of different classes cannot be well separated using a hyperplane.

However, due to the same flexibility, tree classifiers are quite susceptible to overfitting. In ensembles that use tree classifiers, often shallow trees (trees of small levels) are used. In our investigation, we experimented with tree classifiers of different depth as base classifiers and used the performance of the best tree classifier in comparison.

2.3 Regularized Linear Classifiers

Regularization is also a widely used technique in statistics and machine learning to generate better models. Classifiers are trained using training examples. The

training process determines the best parameters for the classifier via optimization, often a minimization of the classifier's prediction error with respect to the training examples. However, in many cases, there are multiple solutions to the optimization problem. And in some other cases, minimizing training error does not lead to the best classifier (e.g., overfitting may happen). Regularization deals with these scenarios following Occam's razor, i.e., the law of parsimony. It adds a preference to the optimization problem such that a simpler solution is preferred. Formally, let w be the parameter (coefficient) vector of a classifier and $L(X, Y, w)$ be a loss function that measures how well the classifier using w can make prediction on the set of training data (X, Y). Rather than training the classifier by minimizing $L(X, Y, w)$, a regularized classifier minimizes:

$$L(X, Y, w) + p(w)$$

where $p(w)$ is a penalty function. The norms of the parameter vector are often used as penalty functions, for example, L2 norm $\|w\|_2 = \sqrt{\sum w_i^2}$ and L1 norm $\|w\|_1 = \sum|w_i|$. (When L2 norm is used as the penalty function, we call the classifier L2-regularized. Same naming scheme applies to L1-regularized classifiers.) To investigate the effectiveness regularization, we experimented with support vector machine, a typical L2-regularized classifier and the L1-regularized logistic regression as base classifiers.

2.3.1 L2-Regularized Linear Classifier: Linear Support Vector Machine

Linear Support Vector Machine (SVM) [13, 14] makes classification using a linear function in the form of:

$$f(x) = wx + b$$

A new data point x is classified into one class if $f(x) \geq 0$ and into the other class if $f(x) < 0$. The parameter vector w and b are obtained by training the SVM using a set of examples. Consider a two-class classification. Suppose the class labels are encoded as "-1" for one class and "+1" for another (i.e., $y^{(i)} \in \{-1, +1\}$). To train an SVM is to perform the optimization:

$$\min_{w,b} \lambda \left\{ \sum_i \left[y^{(i)}\left(wx^{(i)} + b\right) - 1 \right]_+ \right\} + \|w\|_2^2$$

(Eq. 2.1)

where $[\cdot]_+$ is the hinge loss function defined as

$$[x]_+ = \begin{cases} -x & if\ x < 0 \\ 0 & otherwise \end{cases}.$$

Although the formulation is for two-class classification, multiclass classification can be carried out using strategies such as one-vs-one or one-vs-rest. We used linear SVM in our investigation. For each SVM as a base classifier, the hyper parameter λ is determined using cross validation on the training data.

2.3.2 L1-Regularized Linear Classifier: L1-Regularized Logistic Regression

With logistic regression, classification is performed under a probabilistic framework. Suppose there are C classes, i.e., $y^{(i)} \in \{1, 2, \ldots, C\}$. Given a vector x of the gene expression values, the (conditional) probability that the corresponding person belongs to a class t ($t \neq C$) is defined to be

$$\Pr(y = t|x) = \frac{e^{w_t x + b_t}}{Z} \qquad \textbf{(Eq. 2.2)}$$

and for t=C,

$$\Pr(y = C|x) = \frac{1}{Z}$$

where $Z = 1 + \sum_{t=1}^{C-1} e^{w_t x + b_t}$ is a normalizer that ensures the probabilities of the classes for x sum up to one, i.e., $\sum_{t=1}^{C} \Pr(y = t|x) = 1$.

To classify a new data point x, we calculate the probability $\Pr(y = t|x)$ for all the classes $t = 1, 2, \ldots, C$. x is assigned to the class that gives the largest probability.

The parameters (w_t and b_t) are obtained by maximizing the log-likelihood (minimizing the negative log-likelihood) of the training examples, i.e.,

$$\min_{\substack{w_1, w_2, \ldots \\ b_1, b_2, \ldots}} \left\{ -\sum_i \log \Pr(y = y^{(i)}|x^{(i)}) \right\}$$

where $\Pr(y|x)$ follows the definition in Eq. 2.2. The L1-regularized logistic regression adds a regularization term to the optimization:

$$\min_{\substack{w_1, w_2, \ldots \\ b_1, b_2, \ldots}} \lambda \left\{ -\sum_i \log \Pr(y = y^{(i)}|x^{(i)}) \right\} + \sum_t \|w_t\|_1$$

(Eq. 2.3)

Logistic regression is a linear classifier. Consider the two-class classification scenario, the decision boundary is defined by $e^{w_t x + b_t} = 1$, which is a hyperplane in the data space. Also, comparing to L2 regularization, L1

regularization often results in a sparser classifier as the regularization pushes some of the parameters to zero.

We used L1-regularized logistic regression classifier in our investigation. The hyper parameter λ is also determined using cross validation on the training data.

2.4 Evaluation Method

The ensembles were constructed and trained using the same procedure except that the base classifiers are different. We used 10-fold cross validation to evaluate the performance of the different ensembles with different types of base classifiers. Each dataset is randomly partitioned into 10 batches. One batch is put aside as testing data and the rest are used to train the ensemble. The trained ensemble is then tested against the testing data. The average accuracy of the ensemble over the 10 batches is used as the measure of its performance. The same 10-batch partition is used in the evaluation of all the ensembles.

3 Results

3.1 Data and Experimental Settings

Our experiments used publicly available gene expression datasets. Table 3-1 lists the name and the properties of these datasets. We used the Python implementation of the tree classifier provided by Scikit-learn [15]. For SVM, and regularized logistic regression, we used the implementation from Liblinear [16]. Scikit-learn provides a Python interface to Liblinear. The experiments were run on a Linux PC with Intel Core i5 CPU.

Name	Properties
ProstateTumor [17]	This data set contains prostate tumor and normal tissues. There are 102 samples, 10509 genes.
BrainTumor1 [18]	This data set contains five human brain tumor types. There are 90 samples and 5920 genes.
BrainTumor2 [19]	This data set contains four human brain tumor types. There are 50 samples and 10367 genes.
9Tumors [20]	This data set contains nine tumor types. There are 60 samples and 5726 genes.
11Tumors [21]	This data set contains eleven tumor types. There are 174 samples and 12533 genes.

Table 3-1 Names and Properties of the Datasets Used in the Experiments

Except different base classifiers, the ensembles are configured exactly the same. Each ensemble contains 200 base classifiers. The size of the randomly-selected gene subset is set to be the square root of the number of the genes used in the dataset. A preliminary gene selection is applied to the data before they are given to the ensembles.

3.2 Classification Accuracy

Figure 3-1 shows the classification accuracy of the ensembles using different types of base classifiers. Accuracy is measured as the fraction of the correct classification over all test data. The figure plots the average accuracy of 10 rounds of cross validation for each method. Across the experiments, the standard deviation is around 1-2%. Hence we omit the error bar in the plot. We observe that across all the datasets except BrainTumor2 where SVM showed very poor performance, the ensembles that use regularized (both L1 and L2) base classifiers perform better than the one using tree classifier. In particular, with the 11Tumor dataset, the ensembles with regularized base classifiers both achieved classification accuracy around 94% while the ensemble that uses tree classifier only achieved around 86% accuracy.

Across all the datasets, ensemble using L1-regularized logistic regression is consistently the leading performer. With 9Tumor dataset, it reaches around 80% accurate, with the ensemble using SVM second at 75% and the ensemble using classification tree last at only 64%. With BrainTumor2 dataset, it achieves an accuracy of 87% while the ensemble using classification tree achieves 78%. The ensemble using SVM performed poorly and achieved only around 50%.

The results suggest that using a flexible classifier as base classifier may not lead to a better ensemble in many cases. The performance of the ensemble using tree classifier is systematically lower than that of the linear classifiers with regularization. The simple decision boundary employed by the linear classifier does not prevent it from achieving better classification accuracy. Hence the complexity (flexibility) of the base classifier may not be a main concern in designing an ensemble. This answers our first investigation question.

3.3 Effect of Regularization

L1 regularization is often used to construct parsimonious models. By employing the L1 regularization term, the training process (the optimization) removes redundant genes and thus leads to a model that uses fewer genes in classification. In fact, L1-regularized logistic regression has been proposed as a technique for gene selection in classification of gene expression data [22]. It is natural to suspect that the performance advantage displayed by the base classifiers with L1 regularization is due to the possibility that they conducted further gene selection (in an implicit fashion) on the subset of genes involved in the construction of that base classifier.

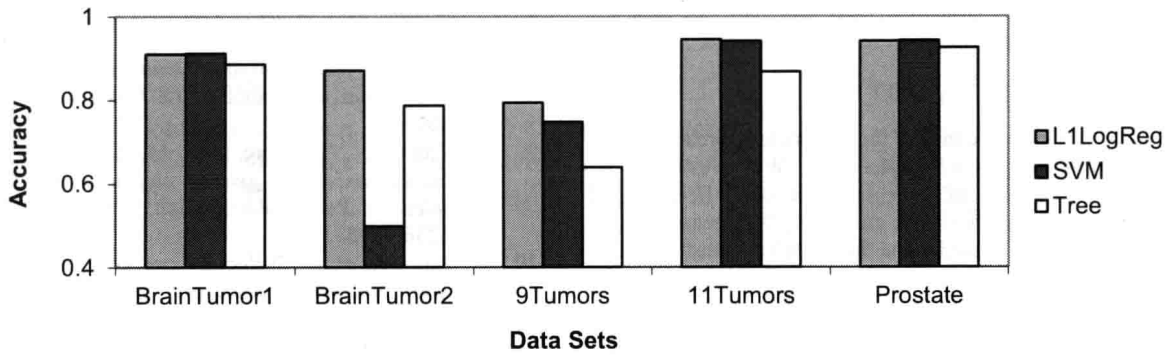

Figure 3-1 Classification Accuracy of the Ensembles Using Different Types of Base Classifiers

To determine whether this is the case, we examined the parameters (coefficients) of the trained base classifiers. If implicit feature selection happened, some of the coefficients would be zero, effectively cancel out the influence of the corresponding genes. To our surprise, the best performing trained base classifiers all have parameter vectors without zero entry. That means all the genes (in the random subset on which the classifier is based on) are included in the model. They all affect the decision of the base classifiers, although with different strength according to the magnitude of the corresponding coefficients. We further manually varied the hyper parameter (λ) of the L1-regularized logistic regression base classifier. Lowering λ increases regularization and the optimization starts to push the coefficients that correspond to some of the (weakly) correlated genes to zero. This effectively removes them from the model. However, we observed that doing so reduces the classification accuracy of the ensemble, even when only a few coefficients are zero (a few genes are removed from the model).

This result suggests that it is not the implicit gene selection but rather the small magnitude of the coefficients caused by the regularization that contributes to the better performance of the L1-regularized base classifier. It answers the second question in our investigation. Regularization can lead to better base classifiers and should be an important direction to explore when designing a good ensemble.

Since both L1 and L2 regularizations generate classifiers with small-magnitude coefficients, we further investigate which type of regularization gives the best performance. From the results presented in Figure 3-1, ensemble based on L1-regularized logistic regression performs better than that based on SVM. However, we notice that L1-regularized logistic regression and SVM differ not only on the type of regularization but also on the loss function (the first term in Eq. 2.1 and the first term in Eq. 2.3). To rule out the possibility that the difference in performance is caused by the different loss functions, we constructed another ensemble that uses L2-regularized logistic regression as base classifier. (It is the same as the L1-regularized logistic regression classifier except that the regularization term, i.e., the second term in Eq. 2.3 becomes $\sum_t \|w_t\|_2^2$.) The performance of this ensemble is very close (almost identical) to that of the ensemble using SVM. It also displayed very poor performance with the dataset BrainTumor2.

This suggests it is L1-regularization that contributes to a better performance. L2-regularization helps but not as well as L1-regularization. In a few cases, L2-regularization may harm classification accuracy.

78	1	0	0	0	8	0	0	3
0	**60**	0	10	0	0	0	0	0
5	12	**47**	1	0	0	5	0	10
2	10	10	**38**	0	0	0	0	0
0	0	0	0	**60**	0	0	0	0
10	0	0	0	0	**70**	0	0	0
8	0	0	0	0	0	**70**	0	2
10	10	0	0	0	0	0	0	0
0	0	1	0	0	1	0	0	**58**

Table 3-2 Confusion Matrix of 9Tumors Using L1LogReg

All datasets used in our experiments except the prostate contain multiple classes. Table 3-2 shows the confusion matrix for the 9Tumors dataset using L1logReg as the base classifier. The table rows represent prediction and columns represent true label. Because 10 rounds of cross validation are performed, the total number of tests is 600 for 60 samples. The classification accuracy is of similar level across different classes except class 8 which is always classified wrong. This is expected as there are

only two examples from that class and they are most likely missed in the training set in cross validation.

4 Conclusions

In this paper, we consider the ensemble approach for classification of gene expression data. We investigated how different types of base classifier can affect the classification accuracy of the ensemble. We focused on the flexibility of a classifier and the classifiers constructed utilizing regularizations. 10-fold cross validation was used to evaluate the classification performance of the ensembles employing different types of base classifiers. The results suggest that to design a good ensemble, classifier complexity is not a main concern but regularization techniques are worth exploring. In particular, base classifiers with L1-regularization may lead to an ensemble of better classification performance.

We limited our consideration to ensembles constructed using random subsets of genes. As a future work, we plan to investigate the impact of base classifiers on ensembles using gene subsets constructed by methods such as genetic algorithm or grouping based on information theory.

5 References

1. Golub, T.R., et al., *Molecular classification of cancer: class discovery and class prediction by gene expression monitoring.* science, 1999. **286**(5439): p. 531-537.
2. Alexandridis, R., S. Lin, and M. Irwin, *Class discovery and classification of tumor samples using mixture modeling of gene expression data—a unified approach.* Bioinformatics, 2004. **20**(16): p. 2545--2552.
3. Dudoit, S., J. Fridlyand, and T.P. Speed, *Comparison of discrimination methods for the classification of tumors using gene expression data.* Journal of the American statistical association, 2002. **97**(457): p. 77-87.
4. Boulesteix, A.L., et al., *Evaluating microarray-based classifiers: an overview.* Cancer Informatics, 2008. **6**: p. 77.
5. Dupuy, A. and R.M. Simon, *Critical review of published microarray studies for cancer outcome and guidelines on statistical analysis and reporting.* Journal of the National Cancer Institute, 2007. **99**(2): p. 147-157.
6. Larrañaga, P., et al., *Machine learning in bioinformatics.* Brief Bioinform, 2006. **7**(1): p. 86-112.
7. Ahn, H., et al., *Classification by ensembles from random partitions.* Journal of Computational Statistics and Data Analysis 2007. **51**.
8. Moon, H., et al., *Ensemble methods for classification of patients for personalized medicine with high-dimensional data.* Artificial Intelligence in Medicine, 2007. **41**(3): p. 197-207.
9. Zhu, Z., Y.S. Ong, and M. Dash, *Markov blanket-embedded genetic algorithm for gene selection.* Pattern Recognition, 2007. **40**(11): p. 3236-3248.
10. Liu, J.J., et al., *Multiclass cancer classification and biomarker discovery using GA-based Algorithms.* Bioinformatics, 2005. **21**(11): p. 2691-2697.
11. Liu, H.W., L. Liua, and H.J. Zhang, *Ensemble gene selection by grouping for microarray data classification.* Journal of Biomedical Informatics, 2010. **43**(1): p. 81-87.
12. Yeh, J.Y., *Applying data mining techniques for cancer classification on gene expression data.* Cybernetics and Systems, 2008. **39**(6): p. 583-602.
13. Burges, C.J.C., *A Tutorial on Support Vector Machines for Pattern Recognition.* Knowledge Discovery and Data Mining, 1998. **2**(4): p. 121-167.
14. Vapnik, V., *The Nature of Statistical Learning Theory*. 1995: Springer-Verlag.
15. *Scikit-Learn.* Available from: http://scikit-learn.org/.
16. Fan, R.-E., et al., *LIBLINEAR: A library for large linear classification.* Machine Learning Research, 2008. **9**: p. 1871-1874.
17. Singh, D., et al., *Gene expression correlates of clinical prostate cancer behavior.* Cancer cell, 2002. **1**(2): p. 203-209.
18. Pomeroy, S.L., et al., *Prediction of central nervous system embryonal tumour outcome based on gene expression.* Nature, 2002. **415**(6870): p. 436-442.
19. Nutt, C.L., et al., *Gene expression-based classification of malignant gliomas correlates better with survival than histological classification.* Cancer Research, 2003. **63**(7): p. 1602-1607.
20. Staunton, J.E., et al., *Chemosensitivity prediction by transcriptional profiling.* Proceedings of the National Academy of Sciences, 2001. **98**(19): p. 10787-10792.
21. Su, A.I., et al., *Molecular classification of human carcinomas by use of gene expression signatures.* Cancer research, 2001. **61**(20): p. 7388-7393.
22. Liao, J.G. and K.-V. Chin, *Logistic regression for disease classification using microarray data.* Bioinformatics, 2007. **23**(15): p. 1945-1951.

Inferring Directed-graph Patterns of Gene Responses to Treatments

Vinhthuy Phan*
Dept. of Computer Science
The University of Memphis
Memphis, TN 38152

Nam S Vo
Dept. of Computer Science
The University of Memphis
Memphis, TN 38152

Thomas R Sutter
Dept. of Biological Sciences
The University of Memphis
Memphis, TN 38152

Abstract

The analysis of patterns of gene expression to multiple treatments can be challenging with small sample sizes, because certain responses cannot be statistically ascertained. When response patterns are represented as directed graphs, however, additional information about true response patterns might be inferred. We exploited the relationship between sample size and a graph property known as *contractibility*, and used synthetic replicates to make inference about patterns of gene response. With microarray data from rats' liver cells, we showed that this approach uncovered subtle interactions in response patterns and resulted in more and better functionally enriched gene clusters.

Keywords: gene expression response patterns, sample size, directed graph.

1 Introduction

Gene expression data allow scientists measure how genes respond to different treatments or environmental conditions. Many studies have focused on finding out how genes of interest respond comparatively to multiple treatments or conditions. In these studies, scientists have argued that multiple samples must be gathered to ascertain patterns of response. Due to biological variation of gene response, sufficient sample size is necessary regardless of the underlying technology used to measure gene expression; for instance, microarray [6] or RNA-seq [2]. Typically, a sample size is calculated for all genes. For example, Lin *et al* [7] calculated a sample size so that a given proportion of genes are significantly expressed with 95% probability.

What can we infer about possible true response patterns from those that are observed with a small number of samples? The answer to this question might depend on how such patterns are represented. One can simply represent each gene's expression as a vector of real numbers. While this representation is good for such analysis as clustering, it does not seem reveal much information about how accurate observed response patterns are. In a number of studies, response patterns to multiple treatments are represented as ternary digits. This representation permits annotation of gene lists [10, 11] and prediction of likelihood of observed patterns [5]. Phan et al. [9] introduced directed-graph representation, with which patterns resulted from a lack of samples might be able to separated from the rest. In these approaches, response patterns obtained from multiple comparisons are completely based on observed data; no attempt is made to infer about patterns that are not observed.

In this paper, we exploited the relationship sample size and a graph property known as "contractibility", and used synthetic replicates to make inference about response patterns when they are represented as directed graphs. We validated prediction accuracy under 4 popular models of gene expression distributions, applied this approach to analyze gene expression in rats' liver cells responding to chemo-preventive chemicals, and finally, showed that predicted patterns can uncover more and better functionally enriched gene clusters that would be otherwise missed.

2 Previous Work

In the context of multiple-treatment comparative gene-expression studies, Phan et al. [9] suggested the use of "response graphs" to represent the patterns of response of genes to treatments. Suppose there are K treatments, and each treatment group has R replicates. A response pattern of significantly differentially expressed gene is determined based on how it responds to all $\binom{K}{2}$ treatment pairs. For treatment pair A and B, a statistical procedures such as the Wilcoxon rank-sum test will produce 3 possible outcomes:

A>B: the gene responds more significantly to A than to B if the pairwise test $H_0 : \mu_A = \mu_B$ is rejected in favor of $H_1 : \mu_A \neq \mu_B$, with $\mu_A > \mu_B$.

A<B: the gene responds more significantly to B than to A if the pairwise test $H_0 : \mu_A = \mu_B$ is rejected in favor of $H_1 : \mu_A \neq \mu_B$, with $\mu_A < \mu_B$.

*Corresponding author. Contact: vphan@memphis.edu

A∼B: If H_0 is accepted because either the gene responds identically to A and B, or the number of replicates is insufficient to resolve its true response.

Based no the outcomes of gene g responding to $\binom{K}{2}$ treatment pairs, the response graph G of g is defined as follows. Vertices of G represent the K treatment groups. Edges of G represent the outcomes of how g responds to all treatment pairs. Specifically, the edge $A \to B$ represents the outcome $A > B$; the edge $B \to A$ represents the outcome $A < B$; and having no edges between A and B represents the outcome $A \sim B$. It can be seen that because of the consistency of comparison tests, it is not possible to have both edges $A \to B$ and $B \to A$. Similarly, G has no cycles.

We call G "contractible" if its non-adjacent vertices are *equivalent*, having identical in-neighbors and out-neighbors. Formally, G is **contractible** if for all non-adjacent vertices u and v, $n(u) = n(v)$ and $N(u) = N(v)$, where $n(a) = \{w : (w, a) \in E_G\}$, and $N(a) = \{w : (a, w) \in E_G\}$. This implies that non-adjacent vertices in a contractible graph are equivalent and can be merged to form a complete graph.

Figure 1 shows examples of contractible and non-contractible graphs.

In [9], we observed that with sufficiently many replicates, the linear order of response of the gene to all treatments are detectable. And this linear order is reflected by the *topological* order of vertices in the contracted, directed, acyclic, complete graph of a contractible response graph. This relationship between sample size and the contractibility of response graphs can be summarized as follows:

Proposition 2.1. *As a gene is given more replicates, its response graph is more likely contractible.*

This observation suggests that non-contractible response patterns are unlikely true response patterns. In this paper, we exploit this relationship to infer true response patterns of genes, by augmenting non-contractible patterns with additional synthetical replicates just enough to make them contractible. Synthetic replicates are carefully generated and made sure to fit the distribution of real replicates.

3 Method

A quick analysis of the non-contractible graph in Figure 1a shows that B being non-distinguishable from both A and C and the fact that $A > C$ are probably due to the lack of samples; as was concluded in a previous work Phan et al. [9]. Further, we see that response graphs (b), (c), (d), (e) and (f) are all possible contractible graphs that contain (a) as subgraph. Thus,

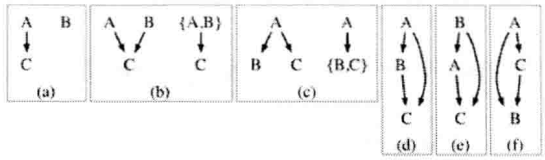

Figure 1: (a) *Non-contractible* response graph of a hypothetical gene with hypothetical 3 treatments. (b-f) five *contractible* response graphs.

one might conclude that the true response must be one of these. But which one?

Given a non-contractible graph G, we generate "synthetic" data for non-adjacent vertices (corresponding to $A \sim B$ outcomes) until a contractible super-graph of G emerges. Synthetic data generation is a probabilistic process, so this step is done multiple times. Consequently, several super-graphs of G can be realized. The most dominant (i.e. most frequent) supergraphs are declared most likely candidates of true responses of the gene. To generate synthetic data, we rely on recent literatures that shed light on possible underlying distributions of gene expression. Bengtsson et al. [1] showed that certain mouse genes were log normally distributed. Hebenstreit et al. [3] studied RNA-seq data and found that the expression levels of a majority of genes in metazoan cells followed a normal distribution or a combination of normal distributions. Based on these works, we used normal distributions as the underlying distributions of gene expressions. Additionally, we used a goodness-of-fit test (2-sample Kolmogorov-Smirnov test) to maintain that synthetic and experimental data likely come from the same distributions. We will show that this simple model can produce accurate predictions.

This probabilistic strategy of inferring true responses of genes to multiple treatments is described in Algorithm 1 .

If a response graph is already contractible, no inference is made. For a non-contractible response pattern, let P be the multi-set of all contractible graphs generated by Algorithm 2 as predictions of true responses of a gene g. These graphs contain the original non-contractible response graph (without synthetic replicates) of gene g as subgraph. Let f_i be the frequency of each $P_i \in P$, and $p_i = \frac{f_i}{|P|}$ be the probability of observing P_i. The entropy of P, $H(P)$ given as $\sum_{P_i \in P} -p_i \log_2 p_i$. $H(P)$ varies between 0 and $\log_2 |P|$. When $H(P) = 0$, P is most certain, consisting of one unique pattern. When $H(P) = \log_2 |P|$, P are uniformly random. Higher entropy implies less certainty in prediction, and vice versa.